Non-Negative Matrix Factorization: Parts-Based Representations for Dimensionality Reduction

Ahmed BADI
ahmedbadi905@gmail.com
linkedin.com/in/badi-ahmed

January 2026

Abstract

Non-Negative Matrix Factorization (NMF) is a powerful technique for decomposing high-dimensional non-negative data into interpretable, low-rank factors. Unlike methods such as PCA that allow positive and negative components, NMF enforces non-negativity in all factors, leading to additive, parts-based representations. This makes NMF especially attractive for domains where negative values are meaningless, such as images, audio spectrograms, text counts, and bioinformatics data. In this article, we introduce the core ideas behind NMF, derive the standard optimization objective, present the classic multiplicative update rules, and discuss common variants and applications. We also compare NMF with other dimensionality reduction methods and highlight its advantages and limitations in real-world machine learning workflows.

Keywords: Non-Negative Matrix Factorization, Dimensionality Reduction, Parts-Based Representation, Topic Modeling, Matrix Decomposition.

1 Introduction

Many types of data are naturally non-negative: pixel intensities in images, word counts in documents, chemical concentrations, and magnitude spectra in audio. When we want to decompose such data into latent factors, it is often desirable that the factors themselves remain non-negative. [1], [2]

Traditional techniques like Principal Component Analysis (PCA) and Singular Value Decomposition (SVD) factorize a data matrix into components that can have both positive and negative entries. These methods are powerful but can be hard to interpret, because they rely on subtractive combinations of basis vectors. [3]

Non-Negative Matrix Factorization (NMF) takes a different approach. It approximates a non-negative data matrix V as a product of two non-negative matrices W and H:

$$V \approx WH,$$

with all entries $V_{ij}, W_{ik}, H_{kj} \geq 0$. The non-negativity constraint leads to purely additive combinations, often resulting in interpretable "parts-based" representations. [1], [4]

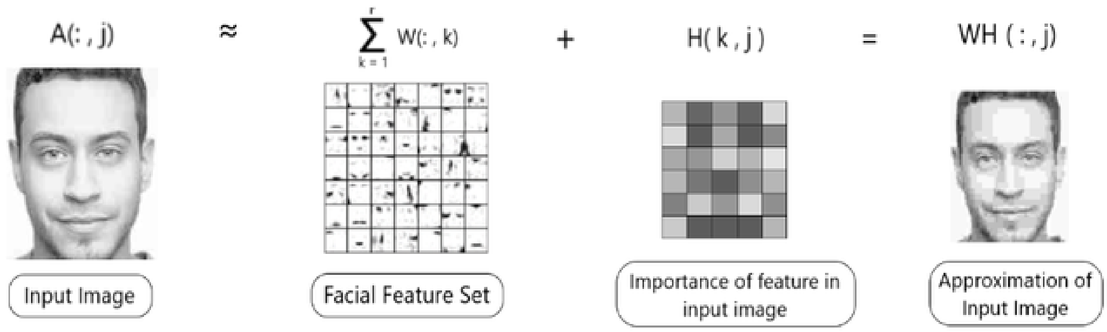

Figure 1: Illustration of NMF learning parts of faces (e.g., eyes, nose, mouth) from face images.

Source: GeeksforGeeks — Non-Negative Matrix Factorization example

In this article, we first formalize the NMF problem, then derive the standard optimization objective and update rules, and finally explore practical applications and variants.

2 Problem Definition

Let $V \in \mathbb{R}_{\geq 0}^{m \times n}$ be a non-negative matrix (for example, m features by n samples). NMF seeks matrices

$$W \in \mathbb{R}_{\geq 0}^{m \times r}, \quad H \in \mathbb{R}_{\geq 0}^{r \times n},$$

such that

$$V \approx WH,$$

where r is the chosen *rank* or number of components, with $r \ll \min(m, n)$. [5]

The columns of W can be interpreted as **basis vectors** or **parts**, and the rows of H as **activation patterns** or **coefficients** that combine these parts to reconstruct each column of V.

For column j, we have

$$V_{:,j} \approx \sum_{k=1}^{r} W_{:,k} H_{k,j},$$

a non-negative linear combination of basis vectors.

3 Objective Function

To find W and H, we minimize a loss that measures how well WH approximates V. Two common choices are: [2], [6]

3.1 Frobenius Norm (Squared Euclidean Distance)

The most common NMF objective uses the Frobenius norm:

$$\min_{W \geq 0,\, H \geq 0} \frac{1}{2} \|V - WH\|_F^2,$$

where

$$\|A\|_F^2 = \sum_{i,j} A_{ij}^2.$$

This objective encourages WH to be close to V in a least-squares sense.

3.2 Kullback–Leibler (KL) Divergence

When V is non-negative and interpreted as counts or intensities, an alternative is the (generalized) KL divergence: [1], [5]

$$D_{\mathrm{KL}}(V\|WH) = \sum_{i,j} \left(V_{ij} \log \frac{V_{ij}}{(WH)_{ij}} - V_{ij} + (WH)_{ij} \right),$$

with the convention $0 \log 0 = 0$.

The NMF problem becomes

$$\min_{W\geq 0,\, H\geq 0} \; D_{\mathrm{KL}}(V\|WH).$$

Different divergences lead to different behavior and are suitable for different data types.

4 Multiplicative Update Rules

Lee and Seung (1999, 2001) proposed simple multiplicative update rules that iteratively minimize the NMF objective while preserving non-negativity. [1], [2]

4.1 Updates for Frobenius Norm

For the Frobenius objective $\frac{1}{2}\|V - WH\|_F^2$, the gradient with respect to H is

$$\frac{\partial}{\partial H} \left(\frac{1}{2}\|V - WH\|_F^2 \right) = W^\top WH - W^\top V.$$

The multiplicative update rule for H is

$$H_{kj} \leftarrow H_{kj} \cdot \frac{(W^\top V)_{kj}}{(W^\top WH)_{kj}},$$

and similarly for W:

$$W_{ik} \leftarrow W_{ik} \cdot \frac{(VH^\top)_{ik}}{(WHH^\top)_{ik}}.$$

These updates guarantee that if W and H are initialized non-negative, they remain non-negative. Moreover, they monotonically decrease the objective under mild conditions. [2], [6]

4.2 Updates for KL Divergence

For the generalized KL divergence objective, Lee and Seung derived the updates: [1]

$$H_{kj} \leftarrow H_{kj} \cdot \frac{\sum_i W_{ik} \frac{V_{ij}}{(WH)_{ij}}}{\sum_i W_{ik}},$$

$$W_{ik} \leftarrow W_{ik} \cdot \frac{\sum_j H_{kj} \frac{V_{ij}}{(WH)_{ij}}}{\sum_j H_{kj}}.$$

Modern implementations also use alternative optimization schemes, such as projected gradient methods or coordinate descent, to achieve faster convergence. [4], [7]

5 Interpretation: Parts-Based Representations

A key appeal of NMF is that it often discovers "parts" of objects instead of holistic components. In the classic experiment by Lee and Seung, NMF applied to face images learned localized features such as eyes, noses, and mouths. Each face could be reconstructed as an additive combination of these parts. [1]

Similarly:

- In document-term matrices, NMF can extract topics (sets of words) and their activations in each document.
- In audio spectrograms, NMF can separate instruments or sound sources.
- In hyperspectral images, NMF can identify pure spectral signatures (endmembers). [4], [8]

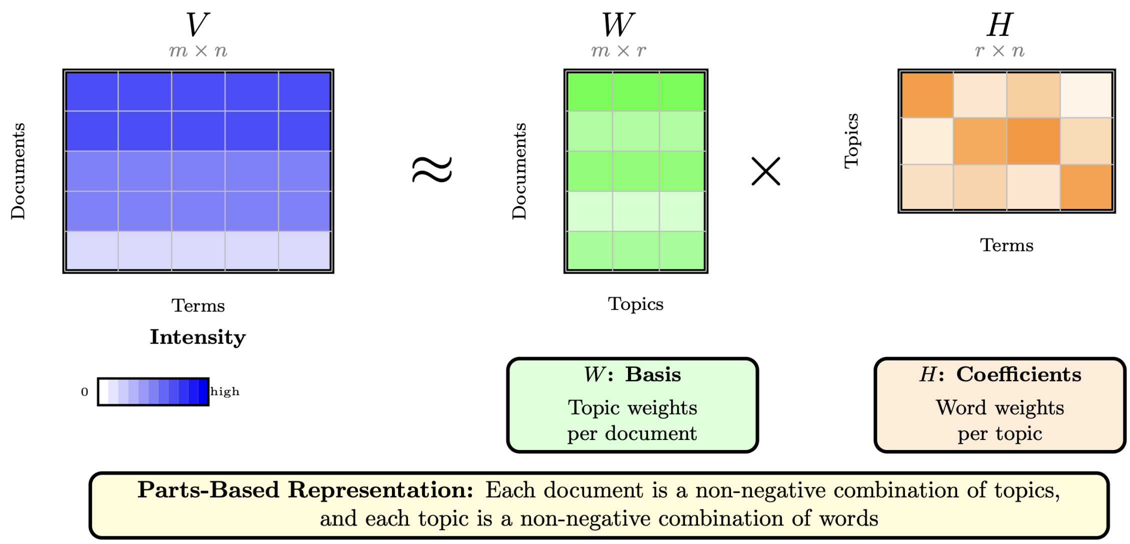

Figure 2: NMF on a document-term matrix: W contains topics (word distributions), H contains document-topic activations.

The non-negativity constraint forces components to represent additive parts, which is more natural for non-negative data than subtractive combination.

6 NMF as Dimensionality Reduction

NMF can be viewed as a dimensionality reduction technique: each sample (column of V) is mapped to a lower-dimensional coordinate vector (column of H). [9]

Given $V \in \mathbb{R}_{\geq 0}^{m\times n}$ and a factorization $V \approx WH$ with $W \in \mathbb{R}_{\geq 0}^{m\times r}$, $H \in \mathbb{R}_{\geq 0}^{r\times n}$, the columns of H are representations in an r-dimensional latent space. These latent features can be used as inputs to clustering or classification algorithms, similar to PCA components but with non-negativity and often better interpretability. [9], [10]

7 Regularized and Variants of NMF

Basic NMF can be extended in many ways to improve robustness and incorporate prior knowledge. [11]

7.1 Sparsity Constraints

To encourage a sparse representation (few active components per sample), we can add L_1 penalties on H and/or W:

$$\min_{W,H\geq 0} \frac{1}{2}\|V - WH\|_F^2 + \lambda_H\|H\|_1 + \lambda_W\|W\|_1.$$

Sparsity often leads to more localized parts and clearer interpretations.

7.2 Graph-Regularized and Deep NMF

Graph-regularized NMF adds a term that encourages similar samples (in a graph) to have similar low-dimensional representations. Deep NMF stacks multiple layers of NMF (or combines NMF with neural networks) to learn hierarchical features. [11], [12]

8 Applications

NMF has been successfully applied in many domains: [1], [8]

- **Topic modeling:** Factorizing document-term matrices to discover topics and their document loadings.
- **Image analysis:** Decomposing images into meaningful parts (e.g., facial features, textures).
- **Bioinformatics:** Extracting gene expression signatures in genomic data.
- **Audio processing:** Source separation and music transcription from spectrograms.
- **Recommender systems:** Factorizing user-item matrices to discover latent preferences.

9 Comparison with Other Methods

Table 1 compares NMF with PCA and k-means clustering in terms of representation properties. [3], [13]

Method	Representation	Constraints	Interpretability
PCA	Linear combinations of orthogonal components	Orthogonality, can have negative values	Moderate
k-means	Assigns each sample to a prototype	Hard cluster assignments	Low (cluster centroids)
NMF	Additive combination of non-negative parts	Non-negativity of factors	High (parts-based)

Table 1: Comparison of NMF with PCA and k-means.

NMF sits between PCA and clustering: it provides a low-rank factorization like PCA but with a more interpretable parts-based structure, and soft combinations of parts similar to soft clustering. [1], [4]

10 Practical Considerations

When using NMF in practice: [10], [13]

- **Non-negativity:** Ensure that input data is non-negative (e.g., use $\max(x, 0)$ or shift data if appropriate).
- **Initialization:** Different random initializations can lead to different local minima; try multiple runs.
- **Rank selection:** Choose the number of components r by cross-validation, reconstruction error, or interpretability.
- **Scaling:** Scaling features can change the factorization; experiment with simple normalizations.
- **Convergence:** Monitor objective values; multiplicative updates may converge slowly compared to more advanced optimizers.

11 Conclusion

Non-Negative Matrix Factorization provides a flexible and interpretable way to decompose non-negative data into low-rank factors. Its key strengths are:

- Non-negativity constraints that lead to additive, parts-based representations.
- Natural applicability to many domains (images, text, audio, bioinformatics).
- Role as both a dimensionality reduction and a feature extraction technique.

We have described the standard NMF formulation, common loss functions, multiplicative update rules, and several applications and extensions. Despite its non-convex nature and local minima, NMF remains a popular tool in modern machine learning, especially when interpretability of latent factors is important. [9], [11]

References

[1] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[2] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems*, vol. 13, 2001.

[3] SAS Communities, *Non-negative matrix factorization (part 1): Understanding its relationship to pca and svd*, `https://communities.sas.com/t5/SAS-Communities-Library/Non-negative-Matrix-Factorization-Part-1-Understanding-Its/ta-p/973592`, 2025.

[4] A. Ang, *Non-negative matrix factorization(s) - applications, theories and algorithms*, `https://angms.science/doc/NMF/20190820UWaterloo.pdf`, 2019.

[5] Wikipedia contributors, *Non-negative matrix factorization*, `https://en.wikipedia.org/wiki/Non-negative_matrix_factorization`, 2024.

[6] R NMF Project, *Nmf algorithm/updates for frobenius norm*, `https://nmf.r-forge.r-project.org/Frobenius-nmf.html`, 2010.

[7] C.-J. Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural Computation*, vol. 19, no. 10, pp. 2756–2779, 2007.

[8] S. Khan et al., "Feature weighted non-negative matrix factorization," *Pattern Recognition*, 2023.

[9] F. Saberi-Movahed, K. Berahman, R. Sheikhpour, Y. Li, and S. Pan, "Nonnegative matrix factorization in dimensionality reduction: A survey," *arXiv preprint arXiv:2405.03615*, 2024.

[10] scikit-learn developers, *Nmf: Non-negative matrix factorization*, `https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.NMF.html`, 2010.

[11] W.-S. Chen et al., "A survey of deep nonnegative matrix factorization," *Neurocomputing*, vol. 500, pp. 365–379, 2022.

[12] X. Liu et al., *Nonnegative matrix factorization with integrated graph*, `https://dl.acm.org/doi/10.1145/2987378`, 2016.

[13] NMF R-Forge, *Nmf algorithm/updates for frobenius norm*, `https://nmf.r-forge.r-project.org/Frobenius-nmf.html`, 2010.